COMBINING GENE EXPRESSION AND CLINICAL DATA TO INCREASE PERFORMANCE OF PROGNOSTIC BREAST CANCER MODELS

Jana Šilhavá and Pavel Smrž
Faculty of Information Technology, Brno University of Technology, Božetěchova 2, 612 66 Brno, Czech Republic

Keywords: Generalized linear models, Logistic regression, Regularization, Combined data, Gene expression data.

Abstract: Microarray class prediction is an important application of gene expression data in biomedical research. Combining gene expression data with other relevant data may add valuable information and can generate more accurate prognostic predictions. In this paper, we combine gene expression data with clinical data. We use logistic regression models that can be built through various regularized techniques. Generalized linear models enables combining of these models with different structure of data. Our two suggested approaches are evaluated with publicly available breast cancer data sets. Based on the results, our approaches have a positive effect on prediction performances and are not computationally intensive.

SCIENCE AND TECHNOLOGY PUBLICATIONS

1 INTRODUCTION

Microarray class prediction (Amaratunga and Cabrera, 2004) is an important application of gene expression data in biomedical research. Microarray experiments monitor gene expressions associated with different phenotypes. Prediction of prognosis based on different phenotypes is challenging due to relatively small number of samples and high-dimensionality of gene expression data. Combining gene expression data with other relevant data may add valuable information and can generate more accurate predictions.

In this paper, we combine gene expressions with clinical data. Clinical data is heterogeneous and measures various entities (e.g. lymph nodes, tumor size), while gene expression data is homogeneous and measures gene expressions. We assume that the combination of gene expressions with clinical data can involve complementary information, which may yield more accurate (disease outcome) predictions than those obtained based on the use of gene expression or clinical data alone. In literature, there are studies aimed at integrative prediction with gene expression and clinical data, e.g. see (Li, 2006) and (Gevaert et al., 2007). On the other side, redundant and correlated data can have contradictory impact on prediction accuracy.

Methods combining biomedical data can be divided into categories depending on the stage of integration (Azuaje, 2010). We propose an approach that combines data at the stage of late integration, which extends a part of work of (Šilhavá and Smrž, 2010). We use logistic regression models that can be built through various regularized techniques and can be applied to high-dimensional data as well. A key to combining gene expression and clinical data is a framework of generalized linear models (GLMs), which is offered for many statistical models.

Simple logistic regression has been widely used with clinical data in clinical trials to determinate the relationship between variables and outcome and to assess variable significance. Clinical data is usually low-dimensional because gene expression data sets include just a few clinical variables. That is why we use simple logistic regression models with clinical data and regularized logistic regression models with high-dimensional gene expression data.

According to (Li, 2006), the penalized estimation methods for integrative prediction and gene selection are promising but computationally intensive. We experimented with R packages 'mboost', 'glmnet', 'grplasso', 'glmpath' that regularize high-dimensional data with penalties and at the same time these statistical models were developed for fitting in GLM framework. R packages 'mboost' and 'glmnet' performed very well and models fitting were not time-consuming. We built the algorithms from these R packages in our classifiers that combine gene expression and clinical data. In case of R package 'mboost', we use a version of boosting that utilizes componentwise linear least squares (CWLLS) as a base proce-

Šilhavá J. and Smrž P..
COMBINING GENE EXPRESSION AND CLINICAL DATA TO INCREASE PERFORMANCE OF PROGNOSTIC BREAST CANCER MODELS.
DOI: 10.5220/0003881505890594
In *Proceedings of the 4th International Conference on Agents and Artificial Intelligence* (SSML-2012), pages 589-594
ISBN: 978-989-8425-95-9
Copyright © 2012 SCITEPRESS (Science and Technology Publications, Lda.)

dure, that closely corresponds to fitting a logistic regression model (Bühlmann and Hothorn, 2007). R package 'glmnet' is an application of elastic net (Zou and Hastie, 2005), which is a regularization and variable selection method that can include both L_1 and L_2 penalties. The algorithms for fitting GLMs with elastic net penalties were developed by (Friedman et al., 2007), which also described logistic regression model with elastic net penalties. Our approaches that combine gene expression and clinical data improve prediction performances and are not computationally intensive.

The rest of this paper is organized as follows: The relevant models with setting of their parameters and the proposed approaches that combine data are described in Section 2. Section 3 presents evaluation methodology and results via comparative boxplots of breast cancer data sets. It also includes a comparison of execution times of applied approaches. This paper is concluded in Section 4.

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

2 METHODS

Notation: Let $\mathbf{X}$ be $p \times n$ gene expression data matrix with an element x_{ij}, p genes and n samples. Let $\mathbf{Z}$ be $q \times n$ clinical data matrix with an element z_{ij}, q clinical variables and n samples. $\mathbf{y}$ is $n \times 1$ response vector with an element y_i and with ground truth class labels $\mathbf{y} \in \{A, B\}$, where A and B can denote poor and good prognosis. In the following text, the upper indexes X, Z, distinguish from variables with gene expression data, clinical data.

2.1 Generalized Linear Models

GLMs (McCullagh and Nelder, 1989) are a group of statistical models that model the response as a nonlinear function of a linear combination of the predictors. These models are linear in the parameters. The nonlinear function (link) is the relation between the response and the nonlinearly transformed linear combination of the predictors. We employ GLMs in data combining due to nice shared properties such as linearity. GLMs are generalization of normal linear regression models and are characterized by the following features:

1. Linear regression model:

$$\eta_i = \beta_0 + \sum_{j=1}^{q} \beta_j x_{ij} + \varepsilon_i, \tag{1}$$

where $i = 1, \ldots, n$. β are regression coefficients and ε is a random mean-zero error term.

2. The link function:

$$g(y_i) = \eta_i, \tag{2}$$

where g is a link function, $i = 1, \ldots, n$. η_i is a linear predictor. Respectively $y_i = g^{-1}(\eta_i)$, where g^{-1} is an inverse link function.

2.2 Logistic Regression Model

We use linear logistic regression model with clinical data. The linear logistic regression model is an example of GLM, where the response variable y_i is considered as a binomial random variable p_i and the link function is logistic:

$$\eta = \log\left(\frac{p}{1-p}\right). \tag{3}$$

Logistic regression model with clinical data can be described with the following equation:

$$g(y_i) = \eta_i = \beta_0^Z + \sum_{l=1}^{q} \beta_l^Z z_{il}, \tag{4}$$

where $i = 1, \ldots, n$. g is the link function (3). y_i or p_i are outcome probabilities $\mathbb{P}(y_i = A | z_{i1}, \ldots, z_{iq})$.

2.3 Boosting Model

A boosting with componentwise linear least squares (CWLLS) as a base procedure is applied to gene expression data. A linear regression model (1) is considered again. A boosting algorithm is an iterative algorithm that constructs a function $\hat{F}(x)$ by considering the empirical risk $n^{-1}\sum_{i=1}^{n} L(y_i, F(x_i))$. $L(y_i, F(x_i))$ is a loss function that measures how close a fitted value $\hat{F}(x_i)$ comes to the observation y_i. In each iteration, the negative gradient of the loss function is fitted by the base learner. The gradient descent is an optimization algorithm that finds a local minimum of the loss function. The base learner is a simple fitting method which yields as estimated function: $\hat{f}(\cdot) = \hat{f}(\mathbf{X}, \mathbf{r})(\cdot)$, where $\hat{f}(\cdot)$ is an estimate from a base procedure. The response $\mathbf{r}$ is fitted against $\mathbf{x}_1, \ldots, \mathbf{x}_n$.

The functional gradient descent (FGD) boosting algorithm, which has been given by (Friedman, 2001) is as follows (Bühlmann and Hothorn, 2007):

1. Initialize $\hat{F}^{(0)} \equiv \sum_{i=1}^{n} L(y_i, a) \equiv \bar{y}$. Set $m = 0$.
2. Increase m: $m = m + 1$. Compute the negative gradient (also called pseudo response), which is the current resudial vector:
   $r_i = -\frac{\partial}{\partial F} L(y, F)|_{F=\hat{F}^{(m-1)}(x_i)}$
   $r_i = y_i - \hat{F}^{(m-1)}(x_i),\ i = 1, \ldots, n.$

3. Fit the residual vector $(r_1,\ldots,r_n)$ to $(\mathbf{x}_1,\ldots,\mathbf{x}_n)$ by base procedure (e.g. regression) $(\mathbf{x}_i,r_i)_{i=1}^n \xrightarrow{\text{base procedure}} \hat{f}^{(m)}(\cdot)$, where $\hat{f}^{(m)}(\cdot)$ can be viewed as an approximation of the negative gradient vector.
4. Update $\hat{F}^{(m)}(\cdot) = \hat{F}^{(m-1)}(\cdot) + \nu \cdot \hat{f}^{(m)}(\cdot)$, where $0 < \nu < 1$ is a step-length (shrinkage) factor.
5. Iterate steps 2 to 4 until $m = m_{stop}$ for some stopping iteration m_{stop}.

The CWLLS base procedure estimates are defined as:

$$\hat{f}(\mathbf{X},\mathbf{r})(x) = \hat{\beta}_{\hat{s}}\hat{x}_{\hat{s}}, \quad \hat{s} = \arg\min_{1\le j\le p}\sum_{i=1}^{n}(r_i - \beta_j x_{ij})^2,$$

$$\hat{\beta}_j = \frac{\sum_{i=1}^n r_i x_{ij}}{\sum_{i=1}^n (x_{ij})^2}, \quad j = 1,\ldots,p.$$

$\hat{\beta}$ are coefficient estimates. $\hat{s}$ denotes the index of the selected (the best) predictor variable in iteration m. For every iteration m, a linear model fit is obtained.

BinomialBoosting (Bühlmann and Hothorn, 2007), which is the version of boosting that we utilize, use the negative log-likelihood loss function: $L(y,F) = \log_2(1+e^{-2yF})$. It can be shown that this population minimizer has the form (Bühlmann and Hothorn, 2007): $F(x_i) = \frac{1}{2}\log\left(\frac{p}{1-p}\right)$, where p is $\mathbb{P}(y_i = A|x_{i1},\ldots,x_{ip})$ and relates to logit function, which is analogous to logistic regression.

2.4 Elastic Net Model

We also use elastic net (Zou and Hastie, 2005) with gene expression data. The linear regression model (1) is considered again. The elastic net optimizes the following equation with respect to β (Friedman et al., 2010):

$$\hat{\beta}(\lambda) = \arg\min \frac{1}{2n}\sum_{i=1}^{n}(y_i - \sum_{j=1}^{p}\beta_j x_{ij})^2 + \lambda\sum_{j=1}^{p}\mathbb{P}_\alpha(\beta),$$

where: $\mathbb{P}_\alpha(\beta) = (1-\alpha)\frac{1}{2}\|\beta\|_{L_2}^2 + \alpha\|\beta\|_{L_1}$ or $\mathbb{P}_\alpha(\beta) = (1-\alpha)\frac{1}{2}\beta_j^2 + \alpha|\beta_j|$ is the elastic net penalty.

$$\mathbb{P}_\alpha(\beta) = \begin{cases} L_1 \text{ penalty} & \text{if } \alpha = 1, \\ L_2 \text{ penalty} & \text{if } \alpha = 0, \\ \text{elastic net penalty} & \text{if } 0 < \alpha < 1. \end{cases} \quad (5)$$

In our case, elastic net builds logistic regression model with elastic net penalties. The regularized equation is fitted by maximum (binomial) log-likelihood and solved by coordinate descent, see (Friedman et al., 2010). The coordinate update has the form:

$$\hat{\beta}_j \leftarrow \frac{S(\frac{1}{n}\sum_{i=1}^n x_i r_{ij}, \lambda\alpha)}{1+\lambda(1-\lambda)} = \frac{S(\beta_j^*, \lambda\alpha)}{1+\lambda(1-\lambda)}, \quad (6)$$

where r_{ij} is the partial residual $y_i - \hat{y}_{ij}$ for fitting $\hat{\beta}_j$ and $S(\kappa,\gamma)$ is the soft-thresholding operator, which takes care of the lasso contribution to the penalty. More detailed description is given in (Friedman et al., 2007).

A simple description of CCD algorithm for elastic net is as follows (Friedman et al., 2010):
The authors assume that the x_{ij} are standardized: $\sum_{i=1}^n x_{ij} = 0$, $\frac{1}{n}\sum_{i=1}^n x_{ij}^2$.

- Initialize all the $\hat{\beta}_j = 0$.
- Cycle around till convergence and coefficitents stabilize:
  1. Compute the partial residuals: $r_{ij} = y_i - \hat{y}_{ij} = y_i - \sum_{k\neq j} x_{ik}\beta_k$.
  2. Compute the simple least squares coefficient of these residuals on jth predictor: $\beta_j^* = \frac{1}{n}\sum_{i=1}^n x_{ij} r_{ij}$.
  3. Update $\hat{\beta}_j$ by soft-thresholding: $\hat{\beta}_j \leftarrow S(\beta_j^*, \lambda)$, which equals (6).

2.5 Combining Gene Expression and Clinical Data

In GLMs, the linear models are related to the response variable via a link function (2). For binary data, we expect that the responses y_i come from binomial distribution. Therefore, logit link function is used in all models with clinical and gene expression data. η_i is a linear model, which is a linear part of logistic regression and a linear regression model in boosting with CWLLS described in Subsection 2.3. We combine the data by summing the linear predictions of clinical and gene expression data:

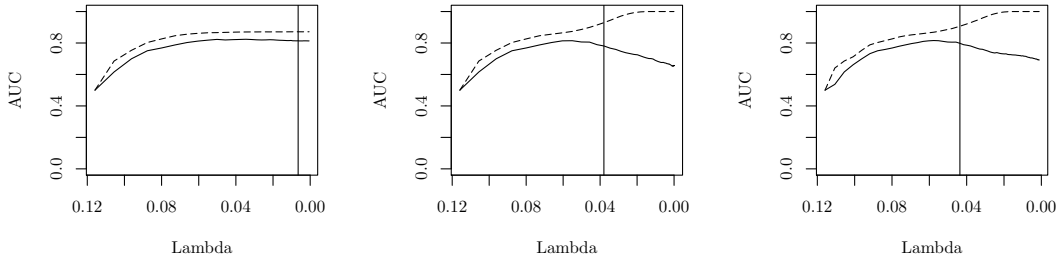
$$\eta_i = \eta_i^Z + \eta_i^X. \quad (7)$$

According to the additivity rule that is valid for linear models, it is possible to sum the linear models:

$$\eta_i = \beta_0^Z + \sum_{l=1}^{q}\beta_l^Z z_{il} + \sum_{j=1}^{p}\beta_j^X x_{ij}. \quad (8)$$

Then the inverse link function g^{-1}, which is the inverse logit function, is applied to the sum of linear predictions η_i:

$$g^{-1}(\eta_i) = \text{logit}^{-1}(\eta_i) = \frac{e^{\eta_i}}{1+e^{\eta_i}}. \quad (9)$$



(a) $p = 20$ (b) $p = 400$ (c) $p = 1000$

Figure 1: Examples of λ solution path produced by glmnet algorithm in the setting with the lasso penalty. The dashed (solid) line denotes AUC estimated from training (test) data set. The vertical line denotes λ_{OPT}.

For better readibility in results, we denote this approach LOG+B.

Similarly we combine logistic regression and regularized logistic regression models from elastic net. For better readability in results, we denote this approach LOG+EN.

3 RESULTS

We tested the described approaches with simulated and publicly available breast cancer data sets. However, the results with simulated data are not shown due to the conference page limit. The performances of individual models were evaluated as well because of a comparison of the models.

In R environment, we used glm function from 'base' package to fit the logistic regression models with clinical data; 'mboost' and 'glmnet' packages to fit the logistic regression models with gene expression data.

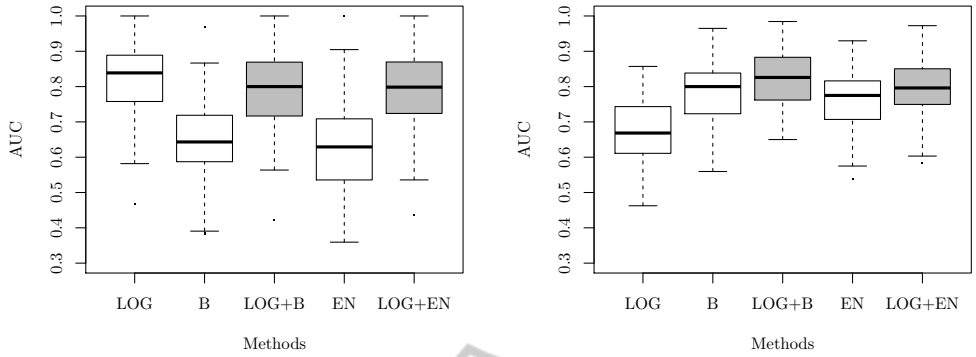
The shrinkage factor ν and the number of iterations of the base procedure are the main tuning parameters of boosting. Based on recommendation from (Bühlmann and Hothorn, 2007), we set $\nu = 0.1$ to the standard default value in 'mboost' package. The numbers of iterations were estimated with Akaike's information stopping criterion (AIC) (Akaike, 1974). We also tested a functionality of AIC stopping criterion, and evaluated performances of data with fixed number of iterations within the range 50-800 iterations, and compared with AIC estimated performances (data not shown). The maximal number of iterations was set to $m_{max} = 700$ and was sufficient.

The choice of the penalty (5) is the main tuning parameter of elastic net. The algorithm from 'glmnet' package computes a group of solutions (regularization path) for a decreasing sequence of values for λ (Friedman et al., 2010). We evaluated all solutions on training and test data set with different penalties α (1, 0, 0.5) and with different numbers of variables (20, 400, 1000) to inspect performances in different dimensional setting. Based on results, we chose the lasso penalty ($\alpha = 1$) for further experiments. Figure 1 depicts examples of this experiment with the lasso penalty. The vertical lines in the subfigures denote the estimated values of λ_{OPT}, which were estimated via training data set cross-validation (CV). The subfigures were generated from simulated gene expression data set of moderate power, see (Šilhavá and Smrž, 2010), and depict one Monte Carlo cross-validation (MCCV) iteration (the same for all figures).

MCCV was applied as a validation strategy. MCCV generates learning set in that way that the learning data sets are drawn out of $\{1, \cdots, n\}$ samples randomly and without replacement. The test data set consists of the remaining samples. The random splitting in non-overlapping learning and test data set was repeated 100-times. The splitting ratio of training and test data set was set to 4 : 1. Responses consisted of predicted class probabilities were measured with the area under the ROC curve (AUC).

We test the described approaches in different settings to simulate various quality of data sets. We considered redundant and non-redundant settings of data and different predictive powers of gene expression and clinical data. From the simulations, the combined models make more accurate predictions or take over the values of the models with higher performances.

We also evaluted the described approaches with two publicly available breast cancer data sets. The van't Veer data set (van't Veer et al., 2002) includes breast cancer patients after curative resection. cDNA Agilent microarray technology was used to give the expression levels of 22483 genes for 78 breast cancer patients. 44 patients that are classified into the good prognosis group, did not suffer from a recurrence during the first 5 years after resection, the remaining 34 patients belong to the the poor prognosis group. The data set was prepared as described in (van't Veer et al., 2002) and is included in R package 'DEN-



(a) van't Veer. (b) Pittman.

Figure 2: Breast cancer data sets. Boxplots of AUCs evaluated over 100 MCCV iterations.

MARKLAB'. The resulting data set includes 4348 genes. Clinical variables are age, tumor grade, estrogen receptor status, progesterone receptor status, tumor size and angioinvasion. The second data set, which is the Pittman data set (Pittman et al., 2004), gives the expression levels of 12625 genes for 158 breast cancer patients. According to recurrence of disease, 63 of these patients are classified into the poor prognosis group, the remaining 95 patients belong to the good prognosis group. Gene expression data was prepared with Affymetrix Human U95Av2 GeneChips. The data was pre-processed using packages 'affy' and 'genefilter' to normalize and filter the data. The genes that showed a low variability across all samples were cleared out. The resulting data set includes 8961 genes. Clinical variables are age, lymph node status, estrogen receptor status, family history, tumor grade and tumor size.

Figure 2 depicts AUC box plots with the breast cancer data sets. Considering the results with the Pittman data set, the combined models have a positive effect on prediction performances and increase AUCs. The combined models, built with the data of van't Veer, do not improve AUC performances and it is better to use for prediction of prognosis clinical data alone. The conclusion with the van't Veer data set also corresponds with findings, e.g. (Gruvberger et al., 2003).

The performances of the combined models are similar. LOG+B seems to perform slightly better than LOG+EN with breast cancer data sets.

Execution Times

The execution times of the combined models are mostly based on the execution times of the models built with high-dimensional data, therefore we compared the execution times of the FGD boosting algorithm from the package 'mboost' (B,LOG+B) with the CCD algorithm from the package 'glmnet' (EN, LOG+EN). Figure 3 depicts the comparison. Increasing numbers of variables are on the horizontal axes, while total execution times for 100 MCCV iterations (in minutes) are on the vertical axes. The plots indicate that both methods achieve similar time values. The execution times grow linearly. Besides, FGD boosting grows with the number of boosting iterations (in our simulations $m_{max=700}$). A grid of 100 λ values is computed in each iteration of EN. The simulations were achieved with a standard PC (Intel T72500 Core 2 Duo 2.00 GHz, 2 GB RAM) and 32-bit operating system.

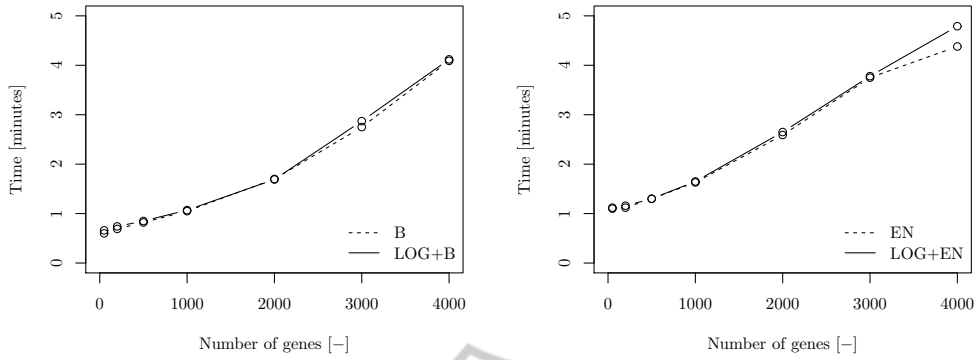
4 CONCLUSIONS

In this paper, we combine gene expression and clinical data to predict disease prognosis. We used logistic regression models built by different ways. GLMs enabled combining of these models. Two suggested approaches were evaluated with simulated (data not shown) and publicly available breast cancer data sets. Both approaches performed well and showed similar performances.

The accuracy of LOG+B has already been compared with other methods from literature in (Šilhavá and Smrž, 2010). It performed the same or better than other methods from literature. LOG+EN can be assessed analogously because of similar prediction performance as LOG+B. The execution times of the combined models grow linearly and the approaches are not time consuming.

ACKNOWLEDGEMENTS

This work was supported by the Technology Agency



(a) B and LOG+B. (b) EN and LOG+EN.

Figure 3: The comparison of computation times and their dependence on increasing number of variables. The graphs are drawn for 100 MCCV iterations.

of the Czech Republic, project TA01010931 – GenEx – System for support of the FISH method evaluation, and the operational programme 'Research and Development for Innovations' in the framework of the IT4Innovations Centre of Excellence project, reg. no. CZ.1.05/1.1.00/02.0070.

REFERENCES

Akaike, H. (1974). *A New Look at the Statistical Model Identification*. IEEE Trans. Automat. Contr., 19 (6), 716-723.

Amaratunga, D. and Cabrera, J. (2004). *Exploration and Analysis of DNA Microarray and Protein Array Data*. John Wiley & Sons, Hoboken.

Azuaje, F. (2010). *Bioinformatics and Biomarker Discovery: "Omic" Data Analysis for Personalized Medicine*. John Wiley & Sons, Singapore.

Bühlmann, P. and Hothorn, T. (2007). *Boosting Algorithms: Regularization, Prediction and Model Fitting*. Statist. Sci., 22, 477-505.

Friedman, J., Hastie, T., Höfling, H., and Tibshirani, R. (2007). *Pathways Coordinate Optimization*. Ann. Appl. Stat., 1, 302-332.

Friedman, J. H. (2001). *Greedy Function Approximation: A Gradient Boosting Machine*. Ann. Statist., 29, 1189-1232.

Friedman, J. H., Hastie, T., and Tibshirani, R. (2010). *Regularization Paths for Generalized Linear Models via Coordinate Descent*. Journal of Statistical Software, 33 (1), 1-24.

Gevaert, O., Smet, F. D., Timmerman, D., Moreau, Y., and Moor, B. D. (2007). *Predicting the Prognosis of Breast Cancer by Integrating Clinical and Microarray Data with Bayesian Networks*. Bioinformatics, 22 (14), 147-157.

Gruvberger, S. K., Ringner, M., and Eden, P. (2003). *Expression Profiling to Predict Outcome in Breast Cancer: the Influence of Sample Selection*. Breast Cancer Res., 5(1), 23-26.

Li, L. (2006). *Survival Prediction of Diffuse Large-B-Cell Lymphoma Based on both Clinical and Gene Expression Information*. Bioinformatics, 22(04), 466-471.

McCullagh, P. and Nelder, J. A. (1989). *Generalized Linear Models*. Chapman and Hall.

Pittman, J., Huang, E., and Dressman, H. (2004). *Integrated Modeling of Clinical and Gene Expression Information for Personalized Prediction of Disease Outcomes*. Proc.Natl.Acad.Sci., 101(22), 8431-8436.

Šilhavá, J. and Smrž, P. (2010). *Improved Disease Outcome Prediction Based on Microarray and Clinical Data Combination and Pre-validation*. Biomedical Engineering Systems and Technologies, 36-41.

van't Veer, L. J., Dai, H., and van de Vijver, M. J. (2002). *Gene Epression Profiling Predicts Clinical Outcome of Breast Cancer*. Nature, 530-536.

Zou, H. and Hastie, T. (2005). *Regularization and Variable Selection via the Elastic Net*. Journal of the Royal Statistical Society, Series B, 67, 301-320.